Security Architecture and Design Documentation Guidance

SECURITY FUNCTIONAL SPECIFICATION

Version 1.0

Prepared by HR CDS TT

23 June 2011

REVISION HISTORY

Name	Date	Reason For Changes	Version
HR CDS TT	23 June 2011	Document creation by Tiger Team	1.0

ACRONYMS AND DEFINITIONS

Acronym	Definition
CCA	Covert Channel Analysis
CDS	Cross Domain Solution
DRD	Development Representation Documentation
DTLS	Descriptive Top-Level Specification
FTLS	Formal Top-Level Specification
HLD	High Level Design
LLD	Low Level Design
SFS	Security Functional Specification
SP	Security Policy

INTRODUCTION

The Security Functional Specification (SFS) describes the system security interfaces and how they function. This consists of all means by which external entities (or subjects in the system but outside of the system's security functions) supply data to the system security functions, receive data from the system security functions, and invoke services from the system security functions.

The goal of the SFS is to describe the security functionality and use from the perspective of an external entity, i.e., how an external entity uses and interacts with the security functions.

Figure 1 shows the relationship of the SFS to the other topic areas described in the DRD. For medium robustness, the FTLS is not present.

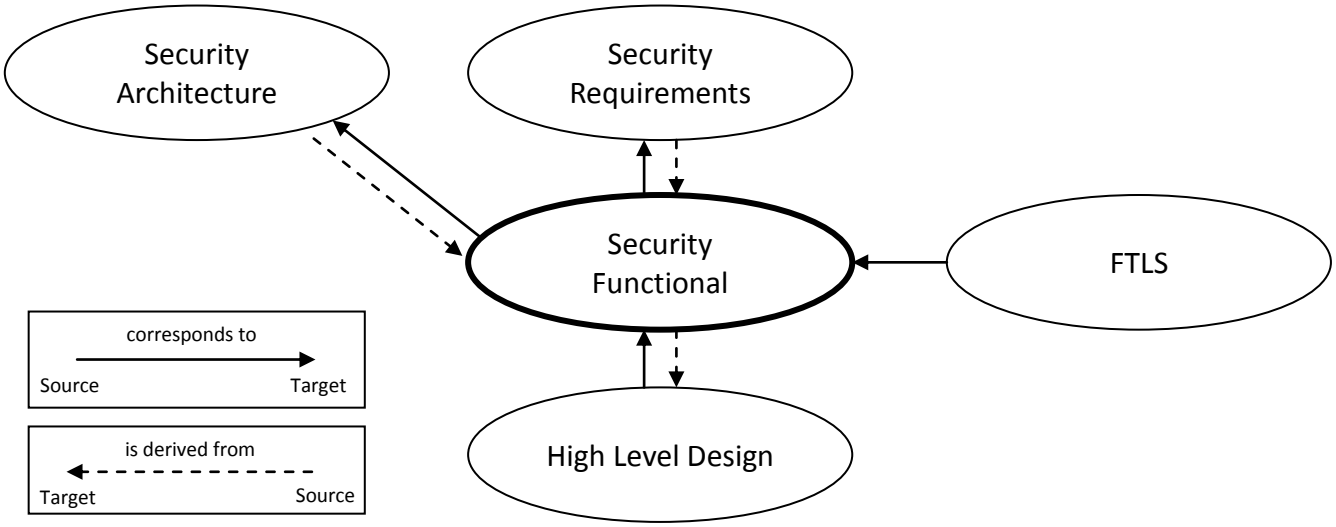

Figure 1 – Security Functional Specification Interactions

DISCUSSION

It is essential that the SFS be complete, correct, understandable, and easy to read in order to provide a collective understanding of the system's security functionality. It is equally important that the SFS identify and define the external interactions and requested behavior with respect to the system's security functions, focusing on what external entities might "observe" when they interact with the system security functions. In this regard, the SFS should describe the security functions in terms of their interfaces, including the means by which external entities invoke associated security services, the outcome of the invocation, and the corresponding responses to those service invocations.

Security functions are likely to expose interfaces that are not security relevant. Categorizing interfaces in a manner consistent with the DRD security function categorization (security enforcing, security supporting, and/or security non-interfering) allows different requirements to be levied upon each interface. This will also allow for a first approximation of where to focus security testing and analysis.

Interface descriptions should include the following details:

- Purpose – A high-level description of the interface that explains its goal (i.e., why one might want to use the interface).
- Method of use – A description of how the interface is used, built around the various interactions available at the interface.
- Parameters – An identification of the explicit inputs to and outputs from the interface that control its behavior.
- Parameter descriptions – A description of the parameters in meaningful terms. For example a valid description of the parameter for interface *createfile(i)*, would be "Parameter i represents the file name and relative path". A description such as "Parameter i is a string", is not sufficient.
- Actions – A description of what the interface *does*, more detailed than purpose which simple describes its goal.
- Error message descriptions – A general description of the interface behavior when error conditions arise.

SECURITY OBJECTIVES

SFS - 1 The developer shall provide a Security Functional Specification (SFS).

SFS - 2 The SFS shall provide traceability to the security requirements.

SFS - 3 The SFS shall completely represent the security policy model or the FTLS based on the level of robustness.

SFS - 4 The SFS shall identify and briefly describe the security functions, categorizing their interfaces as security enforcing, security supporting and security non-interfering with supporting rationale for the categorizations.

SFS - 5 The SFS shall describe for each interface identified in SFS-4:

a. For security enforcing interfaces, the purpose, method of use, parameters, parameter description, actions and error messages.

b. For security supporting interfaces, the purpose, method of use, and parameters.

c. For security non-interfering interfaces, the purpose.